DOCUMENT RESUME

ED 427 663 IR 019 182

AUTHOR Johnson, W. Lewis; Blake, Tyler; Shaw, Erin

TITLE Automated Management and Delivery of Distance Courseware.

SPONS AGENCY Annenberg Foundation, St. Davids, PA.
PUB DATE 1996-10-00
NOTE 9p.; In: WebNet 96 Conference Proceedings (San Francisco, CA, October 15-19, 1996); see IR 019 168. Some figures may not reproduce clearly in the print version.
AVAILABLE FROM Web site: http://aace.virginia.edu/aace/conf/webnet/html/153.htm
PUB TYPE Reports - Descriptive (141) -- Speeches/Meeting Papers (150)
EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS Authoring Aids (Programming); *Computer Assisted Instruction; *Computer Managed Instruction; Computer Mediated Communication; Computer Software Development; *Computer System Design; *Courseware; *Distance Education; Educational Technology; Higher Education; Information Technology; Instructional Design; Instructional Material Evaluation; Optical Data Disks; *World Wide Web
IDENTIFIERS Authorware; University of Southern California; Virtual Classrooms

ABSTRACT

This paper describes a system called ANDES, designed for the management and delivery of distance education courses. ANDES enables students to study at home at their own pace, as well as interact with instructors and other students in virtual classrooms. It uses World Wide Web technology for transmission and delivery, with extensions relevant to distance education. It seamlessly integrates course materials on CD-ROM with materials that have been downloaded over the Internet. Student progress through the course is tracked automatically for evaluation purposes. A high-level authoring language supports rapid development of courseware materials. Custom interfaces for course instructors and administrators provide online views of student work and progress. Topics discussed include: (1) background on use of the Web for courseware delivery; (2) the structure of ANDES courses at the University of Southern California, based on principles of human factors design and applied learning theory--i.e., an emphasis on experiential learning, optional emphasis on mastery, promotion of a class identity through group communication, and variable control of course sequencing; (3) ANDES architecture, including the student mini-server, student workstation organization, and central server organization; (4) processing and communication details; (5) generating Web-based courseware materials; and (6) evaluation, including alpha test results. (Author/DLS)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED 427 663

Automated Management and Delivery of Distance Courseware

W. Lewis Johnson
Tyler Blake
Erin Shaw

Presented at WebNet 96 San Francisco, CA October 15-19, 1996

IR019182

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
☐ This document has been reproduced as received from the person or organization originating it.
☐ Minor changes have been made to improve reproduction quality.
• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

BEST COPY AVAILABLE

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
G.H. Marks
TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

Automated Management and Delivery of Distance Courseware

W. Lewis Johnson, Tyler Blake, and Erin Shaw University of Southern California, U.S.A.
andes@isi.edu

Abstract: This paper describes a system called ANDES for management and delivery of distance education courses. ANDES enables students to study at home, at their own pace, as well as interact with instructors and other students in virtual "classrooms." It uses World Wide Web technology for transmission and delivery, with extensions relevant to distance education. It seamlessly integrates course materials on CD-ROM with materials that have been downloaded over the Internet. Student progress through the course is tracked automatically for evaluation purposes. A high-level authoring language supports rapid development of courseware materials. Custom interfaces for course instructors and administrators provide online views of student work and progress.

1. Background

As the World Wide Web has developed, it has attracted considerable attention as a potential vehicle for courseware delivery. Universities with active distance education programs, such as The Open University [Open 96] and the University of Athabasca [Athabasca 96] are making increasing use of Web. Although the Web usually plays a minor role as an adjunct to conventional materials such as printed textbooks and study guides, there have been experiments using the Web as the primary delivery mechanism, and HTML as the authoring language [Holt et al. 95].

The Web has a number of limitations that stand in the way of effective use for distance education. Accessing pages can be slow, especially by modem; this limits the rate at which students can navigate through material, and effectively precludes the use of large multimedia assets such as video clips. HTML pages are static presentations; making them interactive requires expertise in CGI or Java programming, and is therefore beyond the reach of the typical courseware author. Maintaining networks of Web pages is an error-prone process, as is evidenced by the large number of obsolete links on the Web. HTML is a general-purpose authoring language, and does not commit to any particular look and feel. It therefore provides no built-in support to authors who wish to maintain a consistent look and feel throughout their courseware, or who wish to define templates for common courseware elements. Finally, it is difficult to obtain detailed statistics about student progress through the course material; the log files that Web servers maintain give only a partial picture of student activity.

Research efforts are underway to address some of these limitations. For example, the Hyper-G multimedia system maintains document links in a separate database, which helps prevent links from going stale as documents are modified [Maurer&Lennon 95]. The Intelligent Bandwidth project is developing ways of caching Web pages throughout the network, in order to improve effective bandwidth [Touch 96]. However, these general solutions do not adequately meet the specific needs of distance education. ANDES, in contrast, is designed specifically to support distance education courseware; it achieves dramatic increases in effective bandwidth, and provides specialized capabilities such as authoring support and student tracking. Multiple applications are combined in a single environment, avoiding disruptive shifts in students' attention. These goals are accomplished while adhering to standard communication protocols and employing common Web software.

2. The Structure of ANDES Courses

The School of Cinema-Television at USC is developing the initial courses to be delivered using the ANDES system. Courseware content is delivered via video-taped lectures and CD-ROM-based workshops. The structure of the courses adheres to the following principles of human factors design and

applied learning theory.

1. Emphasis on experiential learning -- Workshops are designed to require students to apply facts and concepts in a creative problem-solving process. For example, if a lecture describes the principles of lighting actors on a film set, a correlated workshop might present the users with a simulated set and continuous controls for light intensity and position, and ask them to produce a specific effect by manipulating the controls.

2. Optional emphasis on mastery -- The workshops can be repeated until satisfactory results are obtained, which is important since experiental learning activities yield a fair number of ``mistakes". However, ANDES can also be used to deliver courses adhering to more traditional competitive approaches.

3. Promotion of a "class" identity through group communication -- Course members are subdivided to small, manageable units (i.e. classes) and assigned a specific teaching assistant who will direct on-line discussions. The units can be created at random or based on instructionally relevant criteria. These classes provide the basis for student interchange, and contribute to the feeling of being involved in group learning activities.

4. Variable control of course sequencing -- A major advantage of remote delivery is that students can follow a self-paced, self-scheduled program. However, if some students advance much faster than others, the interchange among class members will be less effective. Each course and instructor will need to decide how to regulate advancement. The system supports everything from completely self-paced to entirely forced-paced approaches.

All courses incorporate the following student experiences, either in sequence, or with branches and iterations: 1) video presentation of lectures enhanced with broadcast quality production values, 2) interactive workshop exercises requiring application of lecture material, 3) instructor feedback via email, commented workshop screens, and chat group discussions, 4) class discussions of workshops via Internet Relay Chat and/or phone connections, 5) scheduled electronic office hours for students to ask questions of the professor and/or TA, and 6) occasional guest lectures.

ANDES courses are designed by analyzing each educational objective in a weighted consideration of four major factors: relative media effectiveness, time-criticality of the information, cost to implement and the availability of client hardware. From this analysis, the author determines to what platforms to assign the content. For example, in the pilot film course, the quality of CD-ROM-based video is inadequate to depict the production concepts to be taught; thus, the film clips are presented on videotape. Guest lectures need to be timely; thus, they are stored in reduced form (e.g., compressed audio plus still visuals) for smooth streaming to the user's machine.

ANDES Architecture

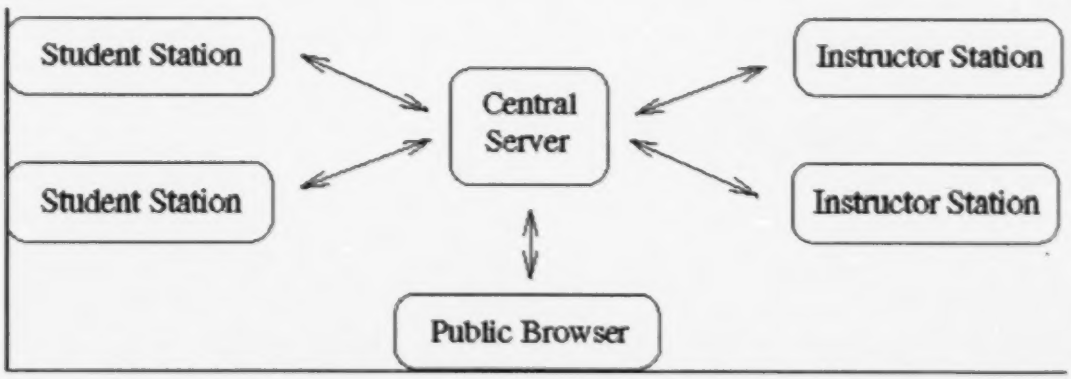

Figure 1: ANDES system configuration.

ANDES is designed to support students, teaching assistants, and course administrators. The roles of the clients and servers in the ANDES architecture are rather different from those in typical client-server architectures. Each student station is able to operate autonomously; when students are working through the course materials on their own, and are not interacting with instructors or other students, no connection to the central servers is required. Connection is established only at the beginning and end of each session, and during live class interactions.

3.1 The Student Mini-Server

Autonomy is accomplished by installing a Web server and CGI program locally on each ANDES student work station. This *student mini-server*, presents course materials to the student as requested, and tracks the student's progress through the course. The mini-server presents the course materials using a conventional Web browser, namely Netscape. The student mini-server contacts the central server automatically when needed, in order to obtain updates for course materials and report on student progress. The mini-server approach provides a number of advantages over the conventional Web as a courseware delivery platform. Storing the course materials locally on the student machine greatly improves system response, and largely eliminates download time during courseware delivery. Autonomous courseware delivery reduces the amount of Internet connect time required, and thus reduces the connect charges to Internet service providers. Because the mini-server resides on the student's machine, it can do a better job of tracking student activities and controlling the user interface than a server residing on a remote machine can. Because the mini-server pre-fetches all lesson materials at once, it does a better job of improving system response than general-purpose mechanisms such as caching.

ANDES refers to all assets by *handles*, or aliases that are resolved at runtime to yield an absolute path and file name. Handles enable courseware developers to replace or supplement assets oyer the life of a course. A global dictionary is used to store the actual pathnames. This approach allows courseware to be updated at any time, by downloading the updates from the central server and updating the handle dictionary. It also permits performance tuning of the courseware, e.g., copying CD-ROM assets temporarily to hard disk.

3.2 Student Workstation Organization

Fig. 2 shows the architecture of the student workstation. It includes a Web browser (Netscape), a Web server (MacHTTP or WINhttpd), and several additional processes for retrieving, displaying, and managing courseware.

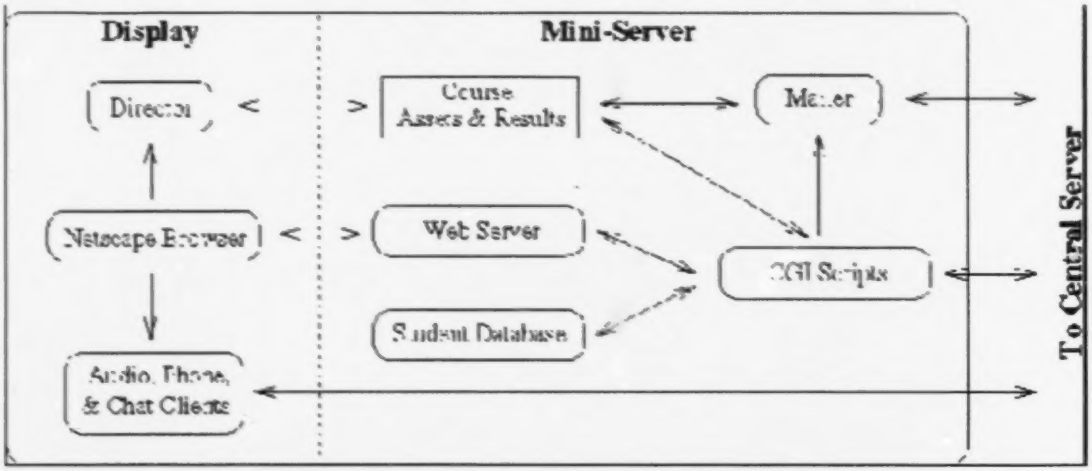

Figure 2: Student workstation architecture.

The Netscape browser communicates with a number of helper applications in order to present

courseware to the student. A customized application is used to startup workshops that have been authored in Macromedia Director. The mini-server currently runs these workshops as separate, full-screen external applications because the Netscape plug-in for Director movies, Shockwave, does not yet support local file I/O. Other helper applications manage chat sessions, play audio transmissions (both live and recorded) and handle two-way voice conversations. The user interfaces of the applications are being adapted and simplified for use specifically within the context of an ANDES course.

The mini-server performs a number of functions, all implemented by scripts employing the Common Gateway Interface (CGI) protocol. It notifies the central server when a student logs in or out of the system. It locates and selects the most up-to-date versions of courseware assets for presentation. It processes high-level descriptions of course screens, written in ANDES Text Markup Language (ATML), and translates them into HTML presentations for display in the Netscape browser. It updates a database, which includes information about the course materials (when they were last updated) and about the student (how far he or she has progressed through the material). It maintains a log of student actions, both those taken while interacting with the Netscape browser and those taken while playing Director movies, and automatically transmits the log to the central server.

The student mini-server and ANDES central servers communicate using a combination of two-way Internet connections (e.g., when the student starts a lesson) and electronic mail (e.g., to transmit courseware updates). A mailer running on the student machine then filters out messages containing courseware updates, and saves the updates on the student's hard disk. Electronic mail is also used for transmitting student work to the central server. The use of electronic mail reduces the amount of Internet connect time required, and reduces the burden on the central servers. This helps to ensure that ANDES can be easily scaled to handle large course sizes.

3.3 Central Server Organization

Fig. 3 shows the organization of the ANDES central servers, which are used to support the various service requests from the student workstations. The central servers can be divided into two groups: those that manage interpersonal communication (chat, audio, and phone servers), and those that manage and distribute course materials. The focus of development in ANDES has been on the latter.

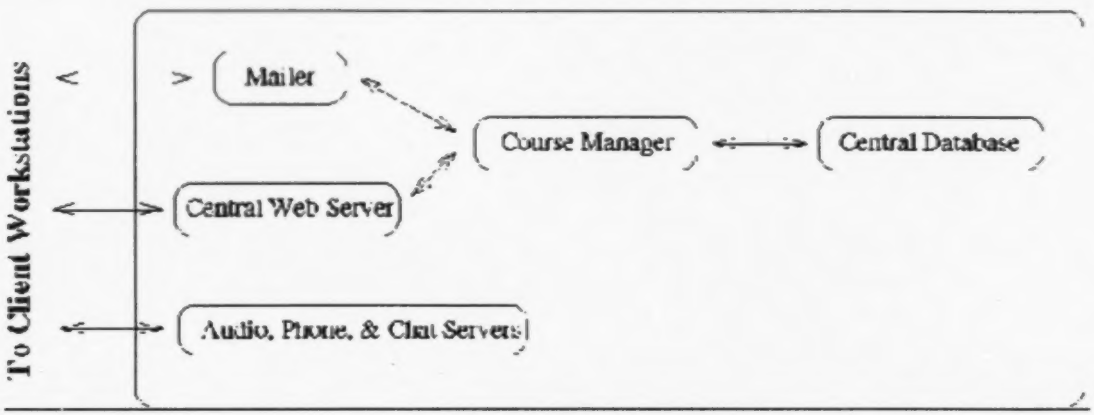

Figure 3: Central server organization.

When a student attempts to log in to the course, or when a student mini-server requests updates of course assets, the request is sent to the *course manager*. The course manager is responsible for validating the request for security purposes, determining what information is required by the client, and delivering the information. The course manager also handles distribution of courseware updates, and requests for information about a student's progress. If a lesson is updated, the course manager determines which student workstations require updates. It is sensitive to the students' progress through the course; if an instructor updates a workshop that a student has already completed, the course manager will inform the mini-server that it is necessary to repeat the workshop.

The Central Web server serves as the course manager's interface to instructors, course administrators, and the interested public. The Web server invokes a CGI stub which connects to the course manager, relays the requests, and then displays the responses on the client's Web browser. The global state of the course is maintained in an Oracle database, which records student data and courseware organization. Student exercises and logs of student activities are stored in the database as well. The databases residing on the student workstations include a subset of the data maintained in the central database.

4. Processing and Communication Details

All communication between the mini-server and course manager is handled via the ANDES protocol, which is based on the Web's Hypertext Transfer Protocol (HTTP). ANDES requests are embedded in the *path* component of HTTP Uniform Resource Locators (URLs). A request contains the script arguments and other necessary data that are passed to the servers for processing. This approach is similar to that used by other context-sensitive Web-based applications, such as I-Doc [Johnson&Erdem 96] and Dienst [Davis&Krafft 95].

The mini-server sends two kinds of requests: *login* requests and *logout*. These functions synchronize the system and guarantee that students receives the necessary course assets. If the student's computer crashes in the midst a session, before a logout request has been issued, this will be detected the next time a login is attempted.

The login connection is as brief as possible and consists of an authentication and a check for assets. If there are assets to be downloaded, the mini-server will inform the student how long the download will take. The student has the option to continue with the download or to quit and log on later at a more convenient time. The assets may either be transmitted immediately using the HTTP connection, or via electronic mail. When the download is confirmed, the login function closes the TCP connection and reconfigures the student's Web server to run locally. When the system is ready, the student is presented with a main menu or new announcement.

The logout function is invoked when the student exits the system. During logout, data files containing student progress, completed work, event logs and email are uploaded or emailed to the course manager. Contextual information is added to the transferred files to assure the appropriate processing by the course manager.

5. Generating Web-Based Courseware Materials

ANDES is designed both to deliver distance courseware and facilitate the creation of such courseware. A special authoring language, called ATML, was developed to facilitate the generation of Web-based courseware.

Within an ANDES course, Web pages may be used to present new material, control interactive exercises, or deliver examinations. These pages should have a consistent form. For example, all pages may include a standard tool bar for navigating to other pages. Test pages might include a built-in timer widget, which locks the page after time runs out. A foreign language course might include multiple pages introducing vocabulary items, each of which contains standard elements like word definitions, examples of uses, and audio clips of spoken language. Consistency of form is achieved by generating pages from templates. The mini-server is configured for each course with a set of page templates appropriate for that course. Each defines a page layout, contains a set of fields to fill in, and defines the actions that the system should take when the student is done with the page. Each ATML page description identifies the template to be used, and lists the field entries. The field entries are indicated with tagged, SGML-like commands. The mini-server dynamically generates HTML pages from the descriptions, and inserts the appropriate CGI processing commands. Developers can combine ATML with HTML, or use just one or the other. ATML files can be created with any word processor.

As an example, consider a possible template for online examinations. Each exam includes a set of questions and a set of corresponding write-in text fields. The page has a title, a button to press when done, links to help pages, etc. Each exam has a fixed duration - exams submitted after the time expires

will be discounted or rejected. Such complex behavior and layout can be generated automatically from the description shown in Fig. 4.

```
<@template>
<@type=open-exam>
<@title=Exam 1>
<@workshop=1A>
<@duration=60mins>
<@text=Exam on the first five lectures of
the course. You have one hour to complete this exam.>
<@questions=Describe the special effects that
can be used in film making./List and describe
the personnel associated with making a movie./Define
in your own words the term "key lighting.">
<@template>
```

Figure 4: Example ATML Template

6. Evaluation

Three phases of testing are planned for ANDES. In the alpha tests, described below, the emphasis was placed on the functionality of specific features of the system and the particular content included. In the beta tests, emphasis will be placed on the usability of the integrated user interface and the flow of student activities. In the first public course a wide variety of assessment techniques will be used, including analysis of the logs of student actions, correlation of patterns of student behavior with performance and satisfaction, time on task, errors, backward navigation, and so on.

6.1 Alpha Test Results

The alpha test was conducted March. At that time, the main student workstation functions and the login functions of the central server were complete. The evaluation lasted about twelve hours over two days and included a feedback session at the end of the last day. Four users, or "students", were stationed at two separate sites on campus and a teaching assistant was located at a third. The students watched the course video tapes and then logged in and took the workshops when appropriate. Eight workshops were tested. Each day concluded with an hour-long, remote TA session that utilized both Internet Relay Chat and Live Audio broadcasting.

When a student logged in, a connection to the central server was established and the student's workshop status was downloaded from the central database. The students were then presented with the main menu from which the workshops were launched; quitting the workshops took the students back to the main menu. At logout, another connection to the central server was made and student work was automatically emailed to the TA.

The feedback session after the evaluation focused on the content and efficacy of the courseware and systems technology. Specific problems and comments were recorded. The consensus was that the sum of the parts worked as a course, that the technology was easy to use, and that students would be interested in taking these "classes". Of particular interest were the chat sessions: The TA said that posing a question during chat was like asking a question in class and having all the students answer simultaneously; and that time would be needed to come up with a strategy for conducting these sessions. The students thought these sessions were effective but found their flow difficult to get used to at first. A maximum of ten students per session was suggested-only half the size we thought we could support. Most said the LiveAudio broadcast was interesting, but not particularly useful (in actuality it was used very little by the TA), although they liked being able to hear the TA's voice.

While the architectural functionality of the system was transparent to students during the evaluation, its setup was time-consuming and needs to be automated. Trying to configure the system to run in the required eight megabytes of RAM was especially troublesome. Ten megabytes are required just to support the Macintosh OS, Netscape, and Director, so virtual memory was used, which caused problems for some of the applications. The memory requirement will be increased to sixteen megabytes, which is

fast becoming the multimedia standard.

References

[HTTP 96] HTTP Working Group (1996). Hypertext transfer protocol 1.0. Web page available at http://www. w3. org/ hypertext/WWW/ Protocols/HTP1.0/ draft-ietf-http-spec.html.

[Holt etal. 95] P. Holt, C. Fontaine, J. Gismondi, and D. Ramsden (1995). Collaborative learning using guided discovery on the internet. International Conference on Computers in Education, AACE. 206-213

[Johnson&Erdem 96] W.L. Johnson and A. Erdem (1996). Interactive explanation of software systems. Automated Software engineering. 3(4).

[Davis&Krafft 95] Building a production technical report server (1995). Advances in Digital Libraries, Spinger-Verlag, NY.

[Maurer&Lennon] H. Maurer and J. Lennon (1995). Digital libraries as learning and teaching support. International Conference on Computers in Education, 1995, AACE. 28-33

[Touch 96] J. Touch. Intelligent bandwidth (1996). Web page available at http://www.isi.edu/div7/ib/.

[Athabasca 96] Athabasca University (1996). Internet undergraduate courses. Web page available at http://www. athabascau.ca/ html/courses/online.htm.

[Open 96] The Open University (1996). Studying via the internet. Web page available at http://cszx.open.ac.uk/zx/.

Acknowledgements

The authors wish to thank the following individuals for their contributions to the ANDES project: John Zigrang of JTZ Engineering proposed the distributed server concept, and has contributed significant technical expertise and insight. Dennis Allard of Oceanpark Software assisted with the prototyping and database design, and advised on the PC implementation. Catherine Labore directed the development of the director courseware. Jay Krish, Jonghee Park and Behnam Salemi all contributed to the implementation of the system. This project is funded through a grant from the Annenberg Foundation.